Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) An electronic system, comprising:

a processor;

a network interface controller including a hardware port;

a source-routed virtual switch implemented in software executed by said processor and including a plurality of software-implemented virtual ports, said virtual ports adapted to provide communication between an application running on said processor and said network interface controller; and

an application programming interface ("API") running on said processor and usable by said application to interface with said virtual switch;

wherein said electronic system is an end node in a network and

wherein said source-routed virtual switch is adapted to receive a packet containing routing information, said routing information identifying an application in user application space that accesses said virtual switch.

2. (Original) The electronic system of claim 1 wherein said API includes code that permits an application to register itself with the virtual switch to permit a resource to be assigned to said application.

3. (Original) The electronic system of claim 1 wherein said API includes code that permits an application to register itself with the virtual switch to permit a unique identifier to be assigned to said application.

4. (Original) The electronic system of claim 1 wherein said API further includes code to deregister said application from virtual switch to release a resource that has been assigned for use by said application.

5. (Original) The electronic system of claim 1 wherein said API includes code to permit said application to transmit data through said virtual switch to another application.

6. (Original) The electronic system of claim 1 wherein said API includes code to cause said virtual switch to open a handle and to post a receive buffer on said handle.

7. (Original) The electronic system of claim 6 wherein said code that causes the virtual switch to open a handle and post a receive also includes code to transition said handle between a first state and a second state, said first state indicates that the switch has not received data to be provided to said application and said second state indicates that the virtual switch has received data to be provided to said application.

8. (Original) The electronic system of claim 6 wherein said API also includes code for closing said handle.

9. (Currently amended) The electronic system of claim 1 wherein said API includes code to permit the application to inform the virtual switch that the application is ready to receive data.

10. (Currently amended) The electronic system of claim 6 wherein said API includes code to permit the application to inform the virtual switch that the application is ready to receive data.

11. (Currently amended) A network, comprising:

a plurality of end nodes; and

at least one switch coupling the nodes together;

wherein at least one of said end nodes includes:

a processor;

a network interface controller including a hardware port;

a virtual switch implemented in software executed by said processor and including a plurality of virtual ports, said virtual ports adapted to ~~be~~ provide communication between an application running on said processor and said network interface controller; and

an application programming interface ("API") running on said processor and usable by said application to interface with said virtual switch;

wherein said virtual switch is adapted to receive a packet containing routing information, said routing information identifying an application in user application space that accesses said virtual switch.

12. (Original) The network of claim 11 wherein said API includes code that permits an application to register itself with the virtual switch to permit a resource to be assigned to said application.

13. (Original) The network of claim 11 wherein said API includes code that permits an application to register itself with the virtual switch to permit a unique identifier to be assigned to said application.

14. (Original) The network of claim 11 wherein said API further includes code to deregister said application from virtual switch to release a resource that has been assigned for use by said application.

15. (Original) The network of claim 11 wherein said API includes code to permit said application to transmit data through said virtual switch to another application.

16. (Original) The network of claim 11 wherein said API includes code to cause said virtual switch to open a handle and to post a receive buffer on said handle.

17. (Original) The network of claim 16 wherein said code that causes the virtual switch to open a handle and post a receive also includes code to transition said handle between a first state and a second state, said first state indicates that the switch has not received data to be provided to said application and said second state indicates that the virtual switch has received data to be provided to said application.

18. (Original) The network of claim 16 wherein said API also includes code for closing said handle.

19. (Currently amended) The network of claim **[[**19**]]** 11 wherein said API includes code to permit the application to inform the virtual switch that the application is ready to receive data.

20. (Currently amended) The network of claim 16 wherein said API includes code to permit the application to inform the virtual switch that the application is ready to receive data.

21. (Currently amended) A computer readable storage medium storing instructions that when executed by a processor cause the processor to implement an application programming interface for a source routed, virtual switch implemented in an end node of a network, said virtual switch implemented in software on a computer, said instructions comprising:

an instruction usable to allocate a resource to an application to permit said application to access said software-implemented, source routed, virtual switch;

an instruction usable to permit said application to transmit data through said software-implemented, source routed, virtual switch; and

an instruction usable to receive data provided to said application through said software-implemented, source routed virtual switch;

an instruction usable for the virtual switch to receive a packet containing routing information, said routing information identifying an application in user application space.

22. (Original) The storage medium of claim 21 further including an instruction usable to deallocate said resource upon said application ceasing using said virtual switch.

23. (Original) The storage medium of claim 21 further including an instruction usable to cause said virtual switch to open a handle and to post a receive buffer on said handle.

24. (Original) The storage medium of claim 23 further including an instruction usable to close said handle.

25. (Previously presented) A method, comprising:

registering an application with a software implemented, source-routed virtual switch implemented in an end node of a network; and

executing an instruction to cause said virtual switch to provide data to said application.

26. (Original) The method of claim 25 wherein registering includes providing a virtual port identifier to said application.

27. (Original) The method of claim 25 wherein registering includes allocating a resource to said application.

28. (Previously presented) The method of claim 25 further including deregistering said application from said software-implemented, source–routed, virtual switch.

29. (Previously presented) The method of claim 25 further including providing an instruction that causes data to be transmitted from said application to said software-implemented, source–routed, virtual switch.

30. (Currently amended) An electronic system, comprising:
a processor;
a network interface controller including a hardware port; and
means for providing an application programming interface to a software-implemented, source-routed switch running on said processor;
means for receiving a packet containing routing information identifying an application in user application space, said application adapted to access said software-implemented, source-routed switch;
wherein said electronic system is an end node in a network.

31. (Previously presented) The electronic system of claim 30 wherein said means includes a means for registering and deregistering said application with said software-implemented, source-routed switch.

32. (Previously presented) The electronic system of claim 30 wherein said means includes a means for sending and receiving data to and from said software-implemented, source-routed switch.

33. (Canceled).

34. (Previously presented) The electronic system of claim 3 wherein said unique identifier assigned to the application is different than an identifier requested for the application.